

**AMENDMENTS TO THE CLAIMS**

This listing of claims will replace all prior versions, and listings, of claims in the application.

**LISTING OF CLAIMS**

1. (Previously amended) A software architecture embodied on a computer-readable medium, the architecture comprising:

multiple attachment points collectively arranged to filter data associated with files that describe software extensions; and

multiple extension managers associated with the multiple attachment points and with respective feature types that can be added to a software platform by software extensions, the extension managers being configured to receive data from the multiple attachment points that pertains only to the feature type with which the extension manager is associated.

2. (Original) The software architecture of claim 1, wherein the attachment points are defined as predicate chains.

3. (Original) The software architecture of claim 1, wherein the attachment points filter XML data.

1 4. (Original) The software architecture of claim 3, wherein each  
2 feature type is associated with an XML tag.

3  
4 5. (Original) The software architecture of claim 3, wherein each  
5 feature type is associated with an XML tag, at least some of the feature types  
6 comprising user-defined feature types.

7  
8 9. (Original) The software architecture of claim 1, wherein each  
9 attachment point exposes collections of ordered nodes.  
10

11  
12 7. (Original) The software architecture of claim 1, wherein each  
13 attachment point exposes collections of ordered XML nodes.  
14

15  
16 8. (Original) A computer embodying the software architecture of  
17 claim 1.

18  
19 9. (Currently amended) A software architecture embodied on a  
20 computer-readable medium, the architecture comprising:

21 a hub structure configured to:

22 receive multiple different files that describe extensions that can be  
23 added to a software platform;

24 combine the multiple different files into a single exposable list; and  
25

1 expose the single exposable list to a filter structure comprising one  
2 or more attachment points ~~that is~~ configured to filter the list.

3  
4 10. (Original) The software architecture of claim 9, wherein the hub  
5 structure receives multiple different XML files and exposes a list of XML nodes.

6  
7 11. (Original) A computer embodying the software architecture of  
8 claim 9.

9  
10  
11 12. (Previously presented) A software architecture embodied on a  
12 computer-readable medium, the architecture comprising multiple different  
13 attachment points each of which is configured to:

14 receive XML data that pertains to one or more software extensions that can  
15 be added to a software platform;

16 process the XML data to provide a list of XML nodes; and

17 expose the list of XML nodes.  
18  
19

20 13. (Original) The software architecture of claim 12, wherein the list  
21 of XML nodes is exposed to another attachment point.  
22  
23  
24  
25

1 14. (Original) The software architecture of claim 12, wherein the list  
2 of XML nodes can pertain to multiple different feature types that can be added by  
3 the one or more software extensions.

4  
5 15. (Original) The software architecture of claim 12, wherein the list  
6 of XML nodes can pertain to multiple different features of particular feature types  
7 that can be added by the one or more software extensions.

8  
9  
10 16. (Original) The software architecture of claim 12, wherein the list  
11 of XML nodes can pertain to one or more of:

12 multiple different feature types that can be added by the one or more  
13 software extensions; and

14 multiple different features of particular feature types that can be added by  
15 the one or more software extensions.  
16

17  
18 17. (Original) A computer embodying the software architecture of  
19 claim 12.

20  
21 18. (Currently amended) A software architecture embodied on a  
22 computer-readable medium, the architecture comprising:

23 a hub structure configured to:  
24  
25

1 receive multiple different files that describe software extensions that  
2 can be added to a software platform;

3 combine the multiple different files into a single exposable list; and  
4 expose the single exposable list to a filter structure that is configured  
5 to filter the list[[:]];

6 [[a]] the filter structure comprising multiple attachment points collectively  
7 arranged to filter data associated with the list exposed by the hub structure; and

8 multiple extension managers associated with the multiple attachment points  
9 and with respective feature types that can be added to a software platform by  
10 software extensions, the extension managers being configured to receive data from  
11 the multiple attachment points that pertains only to the feature type with which the  
12 extension manager is associated.  
13

14  
15 19. (Original) The software architecture of claim 18, wherein the hub  
16 structure receives multiple different XML files and exposes a list of XML nodes.  
17

18  
19 20. (Original) The software architecture of claim 19, wherein the list  
20 contains root node tags for all of the XML files.  
21

22 21. (Original) The software architecture of claim 19, wherein the  
23 XML files logically describe where a particular extension fits on the software  
24 platform.  
25

1  
2 22. (Original) The software architecture of claim 19, wherein the  
3 attachment points are defined as predicate chains.  
4

5 23. (Previously presented) The software architecture of claim 19,  
6 wherein an extension manager is notified whenever an extension comprising a  
7 feature type with which it is associated is added or removed from the software  
8 platform.  
9

10  
11 24. (Original) The software architecture of claim 19, wherein each  
12 feature type is associated with a particular XML tag.  
13

14 25. (Original) A computer embodying the software architecture of  
15 claim 18.  
16

17  
18 26. (Original) A method of providing a software extension  
19 comprising:

20 exposing an XML list that contains one or more nodes;

21 processing the XML list to identify specific nodes that correspond to  
22 various feature types that can be added to a software platform; and  
23

24 notifying an extension manager that is associated with at least one feature  
25 type if a node that corresponds to that feature type is identified in the XML list.

1  
2 27. (Original) The method of claim 26, wherein said processing is  
3 accomplished by filtering the XML list using multiple attachment points that are  
4 defined as predicate chains.  
5

6 28. (Original) The method of claim 27, wherein the individual  
7 attachment points receive XML data as an input and expose a list of XML nodes.  
8  
9

10 29. (Original) The method of claim 26, wherein said processing is  
11 accomplished by filtering on specific nodes.  
12

B1  
13 30. (Previously amended) The method of claim 26, wherein said  
14 processing is accomplished by exposing various nodes.  
15  
16

17 31. (Previously amended) The method of claim 26, wherein said  
18 processing is accomplished by filtering on specific nodes and exposing various  
19 nodes.  
20

21 32. (Original) One or more computer-readable media having  
22 computer-readable instructions thereon which, when executed by a computer,  
23 cause the computer to implement the method of claim 26.  
24  
25

1           33. (Currently amended)           A method of providing a software  
2 extension comprising:

3           receiving XML data that pertains to a software extension that is to be added  
4 to a software platform;

5           processing the XML data using one or more attachment points to identify  
6 XML nodes; and

7           exposing an XML list that contains one or more nodes that are identified by  
8 said processing.  
9

10  
11           34. (Original)   The method of claim 33, wherein said receiving  
12 comprises receiving multiple XML files that pertain to different software  
13 extensions.  
14

15  
16           35. (Original)   The method of claim 34, wherein said processing  
17 comprises combining the multiple XML files into a single exposable list.  
18

19           36. (Currently amended)           The method of claim 33, wherein said  
20 ~~processing comprises processing~~ the XML data with list is exposed to one or more  
21 attachment points that are defined as predicate chains that filter the XML data.  
22

23  
24           37. (Previously amended) The method of claim 36, wherein at least one  
25 of the attachment points exposes a node.



1  
2 38. (Original) The method of claim 36, wherein at least one of the  
3 attachment points filters on a node.  
4

5 39. (Original) One or more computer-readable media having  
6 computer-readable instructions thereon which, when executed by a computer,  
7 cause the computer to implement the method of claim 33.  
8

9  
10 40. (Original) A method of providing a software extension  
11 comprising:

12 receiving multiple different files, each of which being associated with a  
13 different software extension and logically describing its associated software  
14 extension;  
15

16 combining the multiple different files in a single list;

17 exposing portions of the list;

18 processing exposed portions of the list to identify one or more feature types  
19 that are to be added to a software platform; and

20 notifying an extension manager that is associated with a particular feature  
21 type.  
22

23 41. (Original) The method of claim 40, wherein the multiple different  
24 files comprise XML files.  
25

B1

1  
2 42. (Original) One or more computer-readable media having  
3 computer-readable instructions thereon which, when executed by a computer,  
4 cause the computer to implement the method of claim 40.

---

5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25